

**IN THE CLAIMS:**

Please amend the claims to read as indicated herein.

1. (Currently amended) A method, carried out by a status engine, of monitoring services of an information technology (IT) environment, comprising:  
~~based on a~~  
storing a representation of a service hierarchy model, the stored representation  
comprising service elements representing the services of the IT environment ~~by service~~  
~~model elements~~, wherein each service element has an associated service model  
~~statuses are associated with the service model elements~~, the service hierarchy model  
comprising superordinate and subordinate service ~~model elements~~, wherein the status  
of a superordinate service ~~model element~~ depends on at least one of the statuses of  
one or more subordinate service ~~model elements~~; ~~said method comprising:~~
- calculating the status of a superordinate service ~~model element~~ by considering status  
dependency and propagation between service ~~model elements~~ within the service  
hierarchy model according to one or more rules, wherein the rules define the  
dependency of the status of the superordinate service ~~model element~~ on at least one of  
the statuses of one or more subordinate service ~~model elements~~ and the propagation of  
one or more statuses from one or more subordinate service ~~model elements~~ to the  
superordinate service ~~model element~~, the rules comprising at least one of:
- a) a rule that is based on additional attributes of the service ~~model element~~ other  
than the status;
  - b) a rule that ignores subordinate service ~~model elements~~;
  - c) a rule that is defined by a user on the basis of at least one of i) logical and ii)  
arithmetical operations of the status or said additional attributes of subordinate service  
~~model elements~~ ; and
  - d) a rule that is programmed individually by a user.

2. (Currently amended) The method of claim 1, wherein the rules, when the status of the at least one superordinate service ~~model~~-element is calculated, include:

status propagation rules that each have as an input only one parameter, wherein the parameter is the status of the at least one subordinate service ~~model~~-element, and

status calculation rules that have as an input one or more parameters selected from the group consisting of: the propagated status of the at least one subordinate service ~~model~~-elements, messages coming from services of the IT environment, and additional attributes.

3. (Currently amended) The method of claim 1, wherein the calculation of the status of the at least one superordinate service ~~model~~-element depends on any combination of three different types of input data: the status of the at least one subordinate service ~~model~~-element, messages affecting the at least one superordinate service ~~model~~-element and the additional attributes of the service ~~model~~-elements.

4. (Currently amended) The method of claim 1, wherein the additional attributes can take values that are different from possible values of the status of the service ~~model~~ elements.

5. (Currently amended) The method of claim 1, wherein the status of the at least one superordinate service ~~model~~-element is only calculated if certain attributes of the at least one superordinate service ~~model~~-element are set.

6. (Currently amended) The method of claim 1, wherein specific subordinate service ~~model~~-elements of the at least one subordinate service ~~model~~-element are individually treated for the calculation of the status of the at least one superordinate service ~~model~~-element.

7. (Original) The method of claim 1, wherein user-specific external data is included in the rules.

8. (Original) The method of claim 1, wherein time of the day information is included in the rules.

9. (Currently amended) A computer system for monitoring services of an information technology (IT) environment, wherein the computer system monitors the services based on a service ~~model~~hierarchy, wherein a stored representation of the service ~~model~~hierarchy includes service ~~model~~-elements, wherein each of the service ~~model~~-elements ~~represents~~ representing a services of the IT environment and is ~~associated with each having an associated~~ service ~~model~~-status, wherein the service ~~model~~-elements include at least one superordinate service ~~model~~-element and at least one subordinate service ~~model~~-element, wherein a status of the at least one superordinate service ~~model~~-element depends on a status of the at least one subordinate service ~~model~~-element, the system comprising:

a status engine for calculating the status of at least one of the service ~~model~~-elements, wherein the status engine can calculate the status of the at least one superordinate service ~~model~~-element by considering status dependency and propagation between the service ~~model~~-elements within the service ~~model~~hierarchy, according to one or more rules;

a user interface for configuring the rules; and

a graphical display for visualizing monitoring results,

wherein the rules define the dependency of the status of the at least one superordinate service ~~model~~-element on the status of the at least one subordinate service ~~model~~-element and a propagation of the status from the at least one subordinate service ~~model~~-element to the at least one superordinate service ~~model~~-element, and

wherein the rules include at least one of:

- a) a rule that is based on additional attributes of at least one of the service model elements other than the ~~service model~~ status;
  - b) a rule that ignores the at least one subordinate service model element;
  - c) a rule that is defined by a user on the basis of at least one of i) logical and ii) arithmetical operations of the status or the additional attributes of the at least one subordinate service model element; and
- a rule that is programmed individually by a user.

10. (Original) The computer system of claim 9, wherein the interface for configuring the rules is a graphical user interface.

11. (Original) The computer system of claim 9, wherein the interface for configuring the rules is an application programming interface to other programming languages.

12. (Previously presented) The computer system of claim 9, wherein the interface for configuring the rules is a script programming language of which a syntax is provided by the status engine.

13. (Previously presented) The computer system of claim 9, wherein the status engine is capable of handling a graph structure of the IT network of services in which each of the services can have one or more depending services and one or more services on which each of the services depends.

14. (Original) The computer system of claim 9, wherein the dependencies between the services of the IT environment are visualized as a graphical representation.

15. (Currently amended) The computer system of claim 14, wherein the status and status changes of the service ~~model~~-elements are visualized in a graphical representation.

16. (Currently amended) A computer program product including program code, when executed on a computer system, for carrying out, by a status engine, a method for monitoring services within an information technology (IT) environment,

wherein the method ~~is based on~~includes storing a representation of a service ~~model~~hierarchy, wherein the ~~service model includes service model elements, wherein~~ each of the stored representation includes service ~~model~~-elements represents a ~~service~~representing the services of the IT environment and ~~is associated with a each~~ having an associated service ~~model~~-status, wherein the service ~~model~~-elements includehierarchy includes at least one superordinate service ~~model~~-element and at least one subordinate service ~~model~~-element, and wherein ~~the~~a status of the at least one superordinate service ~~model~~-element depends on a status of the at least one subordinate service ~~model~~-element,

wherein the method includes calculating ~~a~~the status of the at least one superordinate service ~~model~~-element by considering status dependency and propagation between the service ~~model~~-elements within the service ~~model~~hierarchy according to one or more rules, wherein the rules define the dependency of the status of the at least one superordinate service ~~model~~-element on the status of the at least one subordinate service ~~model~~-element and a propagation of the status from the at least one subordinate service ~~model~~-element to the at least one superordinate service ~~model~~-element, and

wherein the rules include at least one of:

- a) a rule that is based on additional attributes of at least one of the service ~~model~~-elements other than the ~~service model~~-status;
- b) a rule that ignores the at least one subordinate service ~~model~~-element;

- c) a rule that is defined by a user on the basis of at least one of i) logical and ii) arithmetical operations of the status or additional attributes of the at least one subordinate service model-element; and
- d) a rule that is programmed individually by a user.

17. (Original) The computer program product of claim 16, wherein the program code provides an interface to the user for configuring the rules.

18. (Previously presented) The computer program product of claim 17, wherein the interface for configuring the rules is a graphical user interface.

19. (Previously presented) The computer program product of claim 17, wherein the interface for configuring the rules is an application programming interface to other programming languages.

20. (Previously presented) The computer program product of claim 17, wherein the interface for configuring the rules is a script programming language of which syntax is provided by the status engine.

21. (Currently amended) The method of claim 1, wherein the status of at least one of the service model-elements further depends on one or more messages coming from services of the IT environment and affecting the status of the at least one of the service model-elements and wherein the rules further define the dependency of the status of the at least one of the service model-elements on the messages.

22. (Currently amended) The computer system of claim 9, wherein the status of at least one of the service model-elements further depends on one or more messages coming from services of the IT environment and affecting the status of the at least one of the service model-elements and wherein the rules further define the dependency of the status of the at least one service model-elements on the messages.

23. (Currently amended) The computer program product of claim 16, wherein the status of at least one of the service ~~model-element~~elements further depends on one or more messages coming from services of the IT environment and affecting the status of the at least one of the service ~~model~~-elements and wherein the rules further define the dependency of the status of the at least one of the service ~~model~~-elements on the messages.